The IN Operator

Conditions can contain IN for "element of"

SELECT pname
FROM project
WHERE pnumber IN (1,2,4,5);

SELECT pname
FROM project
WHERE pnumber NOT IN (1,2);

SELECT pname
FROM project
WHERE plocation IN ('Houston', 'Stafford');

The IN Operator

SELECT dependent_name
FROM dependent
WHERE (sex, relationship) IN
    ('M', 'SPOUSE'), ('F', 'DAUGHTER');

Nesting Queries with IN

SELECT lname, fname
FROM employee
WHERE ssn IN
    (SELECT essn
     FROM dependent)
Nesting Queries Examples

- List the Names of all supervisors.
- List the Names of all employees that have a dependent spouse.
- List the SSNs of employees that on some project work the same time as 'John Smith'.
- List departments located in 'Houston'.
- List departments not located in 'Houston'.
- List employees without dependents.
- List projects nobody works on.

The ALL Operator

SELECT fname, lname
FROM employee
WHERE salary > ALL (SELECT salary FROM employee WHERE dno = 5);

Nesting Queries with ALL

- List the names of employees that make the maximum salary.
- List the names of employees that make the minimum salary.
- List the names of employees that make a salary that is different from everybody else's salary.

Naming Scope for nested assignments.
Correlated Nesting Queries

- List the names of employees that have a child of the opposite sex
- List the names of employees that make more money than their supervisors

Existence

Tests that a set is nonempty

\[
\begin{align*}
  &\text{SELECT } \text{fname}, \text{lname} \\
  &\text{FROM employee} \\
  &\text{WHERE EXISTS (SELECT * FROM dependent WHERE ssn = essn)}; \\
  &\text{SELECT } \text{fname}, \text{lname} \\
  &\text{FROM employee} \\
  &\text{WHERE NOT EXISTS (SELECT * FROM dependent WHERE ssn = essn)};
\end{align*}
\]

Set Intersection using EXISTS

Example: Employees with dependents

\[
\begin{align*}
  &\text{SELECT } \text{fname}, \text{lname} \\
  &\text{FROM employee} \\
  &\text{WHERE EXISTS (SELECT * FROM dependent WHERE ssn = essn)}; \\
  &\text{Mangers that are also supervisors}
\end{align*}
\]
Set Difference using NOT EXISTS

Example: Employees without dependents

```
SELECT fname, lname
FROM employee
WHERE NOT EXISTS (SELECT * FROM dependent WHERE ssn = essn);
```

- Employees which are not managers
- Projects on which nobody works 20 hours or more

Unique Existence

Tests that a set contains one element

```
SELECT fname, lname
FROM employee
WHERE UNIQUE (SELECT * FROM dependent WHERE ssn = essn);
```

Not supported by Access or SQL Server

Existence Examples

- List the names of managers that have a dependent
- List the names of employees that work on all department 5 projects [connection to contains]
- List the names of employees that work on all department 4 projects
- List the names of people that work on projects in all departments
Joins in SQL

```sql
SELECT e.*, s.*
FROM employee AS e, employee AS d
WHERE e.superssn = d.ssn;
```

Explicit joins (with variations in Access, SQLServer)

```sql
SELECT e.*, s.*
FROM employee AS e JOIN employee AS d
ON e.superssn = d.ssn;
```

```sql
SELECT e.*, s.*
FROM employee AS e LEFT OUTER JOIN employee AS d
ON e.superssn = d.ssn;
```

Joins in Access

```sql
SELECT employee.*
FROM employee LEFT JOIN dependent
ON employee.superssn = dependent.essn;
```

Similarly right join
No full outer join

Table name necessary

Joins in SQLServer

```sql
SELECT *
FROM employee LEFT JOIN dependent
ON ssn = essn
WHERE relationship = 'daughter';
```

Similarly right join
No full outer join
More Examples

- List employees that work on a research department project, and an administration project.
- List departments which supervise a project that nobody is assigned to work on.
- List all employees and, if they have a supervisor, their supervisor.